

AMENDMENTS TO THE CLAIMS

Claims 1-20, 22-41, and 43-54 are pending. Claim 55 is newly added. Claims 1, 13, 22, 34-36, and 43 are amended.

The listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A computer-implemented method for selectively accessing one or more web services from a client machine, the one or more web services and the client machine being accessible over a network, the method comprising:

providing a first web service description language file describing asynchronous operations for a web service, the first web service description language file describing how to connect to or communicate with the web service using asynchronous communications;

translating the first web service description language file into a second web service description language file describing [[a]]synchronous operations, the second web service description language file describing how to connect to or communicate with the web service using [[a]]synchronous communications;

receiving a request to use [[a]]synchronous communications for communications between the client machine and at least one web service;

providing the second web service description language file to the client machine for generation of client machine code to interact with the at least one web service;

receiving a request for information from the client machine with a conversion engine, the request being received over a synchronous interface;

processing the request in the conversion engine; and

transmitting the processed request over an asynchronous interface from the conversion engine to the at least one web service.

2. (Original) The computer-implemented method of claim 1, wherein the network is one of: a local area network, and a wide area network.

3. (Original) The computer-implemented method of claim 1, further comprising:

receiving a response to the processed request from the at least one web service with the conversion engine, the response being received over the asynchronous interface;

processing the response in the conversion engine; and

transmitting the processed response over the synchronous interface from the conversion engine to the client machine.

4. (Original) The computer-implemented method of claim 1, wherein receiving a request further comprises:

blocking the client machine until one or more of the following events has occurred: a response to the received request has been obtained from the web service and delivered to the client machine, an error message has been delivered to the client machine, and a predetermined time period has passed.

5. (Original) The computer-implemented method of claim 1, wherein processing the request comprises:

receiving the request at a synchronous post interface;
placing the request in a receive queue;
routing the request to one or more delivery queues; and
transferring the request from the delivery queues to one or more asynchronous push interfaces.

6. (Original) The computer-implemented method of claim 1, further comprising:

receiving a confirmation from the at least one web service over the asynchronous interface that the processed request has been received by the at least one web service.

7. (Original) The computer-implemented method of claim 1, wherein transmitting the processed request comprises:

pushing the processed request to the at least one web service over the asynchronous interface.

8. (Original) The computer-implemented method of claim 1, wherein transmitting the processed request comprises:

transmitting an available processed request to the at least one web service through the asynchronous interface in response to polling of the asynchronous interface by the at least one web service.

9. (Original) The computer-implemented method of claim 1, wherein processing the request comprises:

performing security management including one or more of: authentication, authorization, security policy enforcement, decryption, and validation of digital signatures.

10. (Original) The computer-implemented method of claim 2, wherein processing the response comprises:

receiving the response at an asynchronous post interface;
placing the response in a receive queue; and
routing the response to a delivery queue for the client machine.

11. (Original) The computer-implemented method of claim 2, further comprising:

transmitting a confirmation to the at least one web service over the asynchronous interface that the response has been received by the conversion engine.

12. (Original) The computer-implemented method of claim 2, wherein transmitting the processed response comprises:

pushing the processed response to the client machine over the synchronous interface.

13. (Currently Amended) A conversion engine, comprising:
a synchronous interface operable to:

receive a request to interact with one or more web services from a client machine communicating synchronously with the conversion engine over a network, the request generated in accordance with a second web service description language file describing [[a]]synchronous operations for a web service, the second web service description language file describing how to connect to or communicate with the web service using [[a]]synchronous communications, the second web service description language file translated from a first web service description language file describing asynchronous operations, the first web service description language file describing how to connect to or communicate with the web service using asynchronous communications, [[a]]synchronous communications being a requested mode for communications between the client machine and the one or more web services; and

deliver a response to the request from the conversion engine to the client machine over the wide area network;

an asynchronous interface operable to:

deliver the received request from the conversion engine to the one or more web services communicating asynchronously over the wide area network; and

receive a response to the request from the one or more web services over the wide area network; and

a processing module operable to:

convert a synchronous request into an asynchronous request; and

convert an asynchronous response into a synchronous response.

14. (Original) The conversion engine of claim 13, wherein the network is one of: a local area network, and a wide area network.

15. (Original) The conversion engine of claim 13, wherein the asynchronous interface further is operable to:

receive a confirmation from the at least one web service over the asynchronous interface that the processed request has been received by the at least one web service.

16. (Original) The conversion engine of claim 13, wherein the asynchronous interface further is operable to:

transmit a confirmation to the at least one web service over the asynchronous interface that the response has been received by the conversion engine.

17. (Original) The conversion engine of claim 13, further comprising:

a routing module operable to:

route a received request to one or more web services; and

route a received response to the request to the client machine.

18. (Original) The conversion engine of claim 13, further comprising:

a policy directory storing policies for performing security management including one or more of: authentication, authorization, security policy enforcement, decryption, and validation of digital signatures.

19. (Original) The conversion engine of claim 13, further comprising:

a web service directory containing information about available web services and their communication interfaces.

20. (Original) The conversion engine of claim 19, wherein the web service directory includes one or more web service description language files for the available web services.

21. (Canceled).

22. (Currently Amended) The computer-implemented method of claim [[1]] 34, wherein translating includes one or more of the following operations:

translating a types part of the first web service description language file into a types part of the second web service description language file,

translating a message part of the first web service description language file into a message part of the second web service description language file,

translating a port type part of the first web service description language file into a port type part of the second web service description language file,

translating a bindings part of the first web service description language file into a bindings part of the second web service description language file, and

translating a service part of the first web service description language file into a service part of the second web service description language file.

23. (Original) The computer-implemented method of claim 22, wherein translating a types part comprises:

preserving any data structures defined in the first web service description language file in the second web service description language file.

24. (Original) The computer-implemented method of claim 22, wherein translating a types part comprises:

adding an acknowledge element in the asynchronous web service description language file, the acknowledge element describing an acknowledgement that is returned when a request is asynchronously posted to the conversion engine by the client machine.

25. (Original) The computer-implemented method of claim 24, wherein the acknowledgement includes a correlation identifier.

26. (Original) The computer-implemented method of claim 25, wherein the correlation identifier is one or more of: a session identifier, a token, and a call identifier.

27. (Original) The computer-implemented method of claim 22, wherein translating a message part comprises:

adding messages to the asynchronous web service description language file that are particular to asynchronous communication, the messages including one or more of: a message for returning an acknowledgement response, a message for polling, a message for acknowledging a received request, and a message for acknowledging a response from a web service.

28. (Original) The computer-implemented method of claim 27, wherein the message for polling includes one or more of: a message for polling using a session identifier, a message for polling using a topic, and a message for polling using a token.

29. (Original) The computer-implemented method of claim 22, wherein translating a port type part comprises:

inserting a port type for asynchronous post operations and a port type for asynchronous poll operations into the second web service description language file.

30. (Original) The computer-implemented method of claim 29, wherein the port type contains one or more of the following polling options: polling by session identifier, polling by topic, and polling by token.

31. (Original) The computer-implemented method of claim 22, wherein translating a bindings part comprises:

inserting binding for a post port type;

inserting a binding for a poll port type; and

setting an encoding for messages that include the port types to reflect the encoding used by the conversion engine.

32. (Original) The computer-implemented method of claim 22, wherein translating a service part comprises:

adding an asynchronous post port with a first uniform resource locator addressing the conversion engine, and an asynchronous poll port with a second uniform resource locator to the conversion engine.

33. (Original) The computer-implemented method of claim 22, wherein translating comprises:

using a template stored in the conversion engine for translating at least part of the synchronous web service description language file into the asynchronous web service description language file.

34. (Currently Amended) A computer-implemented method for converting a first web service description language file describing [[a]]synchronous operations for a web service into a second web service description language file describing asynchronous operations, comprising:

providing a first web service description language file describing [[a]]synchronous operations for a web service to a conversion engine, the first web service description language file describing how to connect to or communicate with the web service using [[a]]synchronous communications;

translating the first web service description language file in the conversion engine into a second web service description language file describing asynchronous operations, the second web service description language file describing how to connect to or communicate with the web service using asynchronous communications;

receiving a request to use asynchronous communications for communications between the client machine and at least one web service;

providing the second web service description language file to the client machine for generation of client machine code to interact with the at least one web service;

receiving a request for information from the client machine with a conversion engine, the request being received over an asynchronous interface;

processing the request in the conversion engine; and

transmitting the processed request over a synchronous interface from the conversion engine to the at least one web service.

35. (Currently Amended) The computer-implemented method of claim [[34]] 1, wherein translating includes one or more of the following operations:

translating a types part of the first web service description language file into a types part of the second web service description language file,

translating a message part of the first web service description language file into a message part of the second web service description language file,

translating a port part of the first web service description language file into a port part of the second web service description language file,

translating a bindings part of the first web service description language file into a bindings part of the second web service description language file, and

translating a service part of the first web service description language file into a service part of the second web service description language file.

36. (Currently Amended) A computer program product, stored on a machine-readable medium, comprising instructions operable to cause a computer to:

provide a first web service description language file describing asynchronous operations for a web service, the first web service description language file describing how to connect to or communicate with the web service using asynchronous communications;

translate the first web service description language file into a second web service description language file describing [[a]]synchronous operations, the second web service description language file describing how to connect to or communicate with the web service using [[a]]synchronous communications;

receive a request to use [[a]]synchronous communications for communications between the client machine and at least one web service;

provide the second web service description language file to the client machine for generation of client machine code to interact with the at least one web service;

receive a request for information from the client machine with a conversion engine, the request being received over a synchronous interface;

process the request in the conversion engine; and

transmit the processed request over a synchronous interface from the conversion engine to the at least one web service.

37. (Original) The computer program product of claim 36, further comprising instructions to:

receive a response to the processed request from the at least one web service with the conversion engine, the response being received over the asynchronous interface;

process the response in the conversion engine; and

transmit the processed response over the synchronous interface from the conversion engine to the client machine.

38. (Original) The computer program product of claim 36, wherein the instructions to receive a request further comprise instructions to:

block the client machine until one or more of the following events has occurred: a response to the received request has been obtained from the web service and delivered to the client machine, an error message has been delivered to the client machine, and a predetermined time period has passed.

39. (Original) The computer program product of claim 36, wherein the instructions to process the request comprise instructions to:

receive the request at a synchronous post interface;
place the request in a receive queue;
route the request to one or more delivery queues; and
transfer the request from the delivery queues to one or more asynchronous push interfaces.

40. (Original) The computer program product of claim 36, wherein the instructions to process the request comprise instructions to:

perform security management including one or more of: authentication, authorization, security policy enforcement, decryption, and validation of digital signatures.

41. (Original) The computer program product of claim 36, wherein the instruction to process the response comprise instructions to:

receive the response at an asynchronous post interface;
place the response in a receive queue; and
route the response to a delivery queue for the client machine.

42. (Canceled).

43. (Currently Amended) The computer program product of claim [[36]] 55, wherein the instructions to translate include instructions to perform one or more of:

translate a types part of the first web service description language file into a types part of the second web service description language file,

translate a message part of the first web service description language file into a message part of the second web service description language file,

translate a port part of the first web service description language file into a port part of the second web service description language file,

translate a bindings part of the first web service description language file into a bindings part of the second web service description language file, and

translate a service part of the first web service description language file into a service part of the second web service description language file.

44. (Original) The computer program product of claim 43, wherein the instructions to translate a types part comprise instructions to:

preserve any data structures defined in the first web service description language file in the second web service description language file.

45. (Original) The computer program product of claim 43, wherein the instructions to translate a type part comprise instructions to:

add an acknowledge element in the asynchronous web service description language file, the acknowledge element describing an acknowledgement that is returned when a request is asynchronously posted to the conversion engine by the client machine.

46. (Original) The computer program product of claim 45, wherein the acknowledgement includes a correlation identifier.

47. (Original) The computer program product of claim 46, wherein the correlation identifier is one or more of: a session identifier, a token, and a call identifier.

48. (Original) The computer program product of claim 43, wherein the instructions to translate a message part comprise instructions to:

add messages to the asynchronous web service description language file that are particular to asynchronous communication, the messages including one or more of: a message for returning an acknowledgement, a message for polling, a message for acknowledging a received request, and a message for acknowledging a response from a web service.

49. (Original) The computer program product of claim 48, wherein the message for polling includes one or more of: a message for polling using a session identifier, a message for polling using a topic, a message for polling using a token.

50. (Original) The computer program product of claim 43, wherein the instructions to translate a port part comprise instructions to:

insert a post port for asynchronous operation and a poll port for asynchronous operation into the second web service description language file.

51. (Original) The computer program product of claim 50, wherein the port type contains one or more of the following polling options: polling by session identifier, polling by topic, and polling by token.

52. (Original) The computer program product of claim 43, wherein the instructions to translate a bindings part comprise instructions to:

insert a binding for a post port type;

insert a binding for a poll port type; and

set an encoding for the messages that include the port types to reflect the encoding used by the conversion engine.

53. (Original) The computer program product of claim 43, wherein the instructions to translate a service part comprise instructions to:

add an asynchronous post port with a first uniform resource locator addressing the conversion engine, and an asynchronous poll port with a second uniform resource locator to the conversion engine.

54. (Original) The computer program product of claim 43, wherein the instructions to translate comprise instructions to:

use a template stored in the conversion engine for translating at least part of the synchronous web service description language file into the asynchronous web service description language file.

55. (New) A computer program product, stored on a machine-readable medium, comprising instructions operable to cause a computer to:

provide a first web service description language file describing synchronous operations for a web service to a conversion engine, the first web service description language file describing how to connect to or communicate with the web service using synchronous communications;

translate the first web service description language file in the conversion engine into a second web service description language file describing asynchronous operations, the second web service description language file describing how to connect to or communicate with the web service using asynchronous communications;

receive a request to use asynchronous communications for communications between the client machine and at least one web service;

provide the second web service description language file to the client machine for generation of client machine code to interact with the at least one web service;

receive a request for information from the client machine with a conversion engine, the request being received over an asynchronous interface;

process the request in the conversion engine; and

transmit the processed request over a synchronous interface from the conversion engine to the at least one web service.